Augmented Network Embedding in Attributed Graphs

Daokun Zhang

Faculty of Engineering and Information Technology
University of Technology Sydney

This dissertation is submitted for the degree of
Doctor of Philosophy

August 2019

I would like to dedicate this thesis to my loving parents.

Certificate of Original Authorship

I, Daokun Zhang declare that this thesis, is submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the Faculty of Engineering and Information Technology at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise reference or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

This research is supported by the Australian Government Research Training Program.

Production Note:
Signature: Signature removed prior to publication.

Date: 30 Aug, 2019

Acknowledgements

I would like to express my sincere gratitude to my supervisors, Doctor Jie Yin, Professor Xingquan Zhu and Professor Chengqi Zhang, for their selfless devotion to supporting and cultivating me in past three and a half years.

I would like to acknowledge my external supervisor Doctor Jie Yin, who provided me with a valuable opportunity to study at CSIRO (Commonwealth Scientific and Industrial Research Organisation) as a visiting student. As my supervisor, Doctor Yin worked responsibly and patiently to train me in all aspects of my PhD study, from research idea conceiving and technical implementation to paper writing and presentation skill developing. From her, I learned not only the essential skills for independent scientific research, but also the positive attitude as a responsible researcher, which both will constantly benefit me throughout my research career.

I would also like to express my thanks to my principal supervisor Professor Chengqi Zhang, who offered me an unique opportunity to study at the Centre for Artificial Intelligence, UTS, where I was able to network with and learn from so many talented colleagues. From the beginning scholarship application to the final thesis submission, Professor Zhang constantly gave me great support. His warming support cheered me up to behave actively in the face of stress and hardships during my PhD life. His wisdom also inspired me in how to overcome the anxiety caused by peer pressure and try my best to make progress.

I would also like to thank my external supervisor Professor Xingquan Zhu, who encouraged me to pursue PhD study. Before and during my PhD study, Professor Zhu gave me great encouragement to aim high-quality research. Even though through Skype and E-mail, I was so glad to communicate with him to discuss our research work, which was really an encouraging and eye-opening experience. His optimism, and

passion in solving challenging research problems deeply affected me, and will encourage me to make research progress endlessly.

I want to thank my academic brother Doctor Shirui Pan, who recommended me to Professor Xingquan Zhu. During my PhD study, Doctor Pan offered me a lot of generous help, not only in my study but also in my life, especially in the early stage of my PhD life. He provided many valuable suggestions to help me start up my research. He also helped me to adapt to the oversea student life.

I am delighted to thank my friends, colleagues, housemates. We shared our life, laughter and tears, which made my oversea student life much colorful. Because of them, I no longer felt lonely in Australia. They gave me great support and comfort when I was in trouble. I was so glad to know and experience all of them.

I acknowledge the financial support from CSC-UTS Scholarship, CSIRO Top-up Scholarship, CIKM Travel Award (2016), ICDM Travel Award (2016 and 2018). Without these financial supports, I couldn't have finished my PhD study.

Finally, above all, I want to show my special thanks to my family: my parents, my brother, my sister-in-law and my little niece. Because of them, my every effort became meaningful. I especially thank my parents for their self-sacrifice and endless love. They always forgave, understood, supported me unconditionally. To them, I dedicate this dissertation.

Abstract

With the widespread use of information technologies, information networks are becoming increasingly popular to capture complex relationships across various disciplines, such as social networks, citation networks, telecommunication networks, and biological networks. Analyzing these networks sheds light on different aspects of social life, such as the structure of societies, information diffusion, and communication patterns. In reality, however, the large scale of information networks often makes network analytic tasks computationally expensive or intractable. Network embedding has been recently proposed as a new learning paradigm to embed network nodes into a low-dimensional vector space. This facilitates the original network to be easily handled in the new vector space for further analysis. Existing research on network embedding mainly focuses on capturing the structure relatedness in the embedding space, while ignores the important information carried by the widely existing node attributes and labels, which limited the network embedding performance significantly. In this thesis, we dealt with the research problem of augmented network embedding in attributed graphs that aims to learn informative node vector-format representations by augmenting network topology structure with node content attributes and node labels if available. We summarized four research challenges in augmented network embedding: (1) *heterogeneity* caused by the discrepancy between network structure and node attributes/labels; (2) *data sparsity* in network structure and node attributes/labels; (3) *scalability* for handling large-scale networks; (4) *task orientation* for directly benefiting specific network analytic tasks.

To overcome the above challenges, we proposed a series of augmented network embedding algorithms in this thesis. To handle the *heterogeneity* challenge, we proposed the HSCA algorithm that effectively encodes the similarity measured by homophily, structural context and node content attributes into a unified node representation

through the regularized inductive matrix factorization. The attri2vec algorithm was then proposed to address the *heterogeneity* and *data sparsity* challenges, in which node representations are learned by discovering an attribute subspace that better respects network structure. For handling large-scale incomplete networks, we proposed the SINE algorithm that learns node representations by simultaneously modeling node-neighbor and node-attribute relations through a three-layer neural network, with an efficient Stochastic Gradient Descent based online learning strategy. The above three augmented network embedding algorithms only augment network structure with node content attributes, with the purpose to obtain more informative network representations. They are unsupervised, task-general and incapable of directly benefiting specific tasks. To seamlessly integrate network embedding with network analytic tasks, we proposed two task-orientated network embedding algorithms. For collective classification on sparsely labeled networks, we proposed the discriminative attributed network embedding algorithm DMF that integrates network embedding with an empirical loss minimization for classifying node labels, with the purpose of simultaneously exerting the discriminative power of node labels and informativeness of node representations. For searching similar nodes efficiently on large-scale networks, BinaryNE was proposed to learn binary node representations from network structure and node content attributes so that node similarity search can be efficiently done through the fast bitwise Hamming distance calculation performed on the learned binary node representations. To verify the effectiveness of the proposed algorithms, extensive experiments were carried out on nine real-world attributed networks, showing the advantage of the proposed algorithms over state-of-the-art baselines.

Table of contents

List of figures **xiii**

List of tables **xvi**

1 Introduction **1**

1.1 Background and Motivation 1

1.2 Research Challenges 3

1.3 Research Problems 5

1.3.1 Task-general Attributed Network Embedding 5

1.3.2 Task-dependent Attributed Network Embedding 6

1.4 Thesis Contributions 8

1.4.1 Task-general Attributed Network Embedding 8

1.4.2 Task-dependent Attributed Network Embedding 10

1.5 Thesis Overview 11

1.6 Publications 12

2 Literature Review **14**

2.1 Network Embedding 14

2.1.1 Structure Preserving Network Embedding 14

2.1.2 Attributed Network Embedding 18

2.2 Collective Classification 20

2.3 Node Similarity Search 21

3 Preliminaries **23**

3.1 Notations 23

3.2 Definitions 24

3.3 Benchmark Datasets 27

I Task-general Attributed Network Embedding 29

4 Homophily, Structure, and Content Augmented Network Embedding 32

4.1 Introduction 32
4.2 Problem Definition and Preliminaries 35
4.2.1 Problem Definition 36
4.2.2 Preliminaries: Text-Associated DeepWalk 36
4.3 HSCA: Homophily, Structure, and Content Augmented Network Embedding 37
4.3.1 The Optimization Problem 37
4.3.2 Solving the Problem 39
4.4 Experiments 42
4.4.1 Benchmark Networks 43
4.4.2 Experimental Settings 44
4.4.3 Baseline Methods 44
4.4.4 Comparison of Network Representations 45
4.4.5 Convergence Analysis 48
4.4.6 Parameter Sensitivity 48
4.4.7 Visualization of Learned Representations 49
4.5 Conclusion 50
4.6 Appendix 51

5 Attributed Network Embedding via Subspace Discovery 54

5.1 Introduction 54
5.2 Problem Definition and Preliminaries 59
5.2.1 Problem Definition 59
5.2.2 Preliminaries: DeepWalk 60
5.3 The attri2vec Framework 61
5.4 Experiments 65
5.4.1 Benchmark Networks 66

5.4.2 Baseline Methods . . . 67
5.4.3 Experimental Settings . . . 67
5.4.4 Node Classification Experiments . . . 68
5.4.5 Node Clustering Experiments . . . 71
5.4.6 Comparison of Gradient Descent *vs.* Stochastic Gradient Descent 73
5.4.7 Comparison of Shallow *vs.* Deep Mapping . . . 73
5.4.8 Experiments on Out-of-sample Extension . . . 75
5.4.9 Parameter Sensitivity Study . . . 77
5.5 Conclusion . . . 77

6 Scalable Incomplete Network Embedding 80
6.1 Introduction . . . 80
6.2 Problem Definition and Preliminaries . . . 82
6.2.1 Problem Definition . . . 82
6.2.2 Preliminaries: DeepWalk . . . 83
6.3 SINE: Scalable Incomplete Network Embedding . . . 84
6.3.1 Model Architecture . . . 84
6.3.2 Model Optimization . . . 86
6.4 Experiments . . . 89
6.4.1 Benchmark Networks . . . 89
6.4.2 Baseline Methods . . . 90
6.4.3 Experimental Settings . . . 91
6.4.4 Performance Comparison on Incomplete Networks . . . 92
6.4.5 Experiments on Parameter Sensitivity . . . 100
6.4.6 Running Time Comparison . . . 101
6.5 Conclusion . . . 102

II Task-dependent Attributed Network Embedding 103

7 Discriminative Attributed Network Embedding for Collective Classification 105
7.1 Introduction . . . 105
7.2 Problem Definition and Preliminaries . . . 108

7.2.1 Problem Definition . . . 109
7.2.2 Preliminaries: Text-Associated DeepWalk . . . 109
7.3 DMF: Discriminative Matrix Factorization . . . 111
7.3.1 The Optimization Problem . . . 111
7.3.2 Solving the Problem . . . 113
7.3.3 Node Classification Using Learned Representations . . . 117
7.4 Experiments . . . 117
7.4.1 Benchmark Networks . . . 117
7.4.2 Experimental Settings . . . 119
7.4.3 Baseline Methods . . . 119
7.4.4 Comparison of Classification Performance . . . 120
7.4.5 Convergence Analysis . . . 122
7.4.6 Parameter Sensitivity . . . 122
7.4.7 Visualization of Learned Representations . . . 124
7.5 Conclusion . . . 124
7.6 APPENDIX . . . 125

8 Binary Attributed Network Embedding for Efficient Search 128
8.1 Introduction . . . 128
8.2 Problem Definition and Preliminaries . . . 132
8.2.1 Problem Definition . . . 132
8.2.2 Preliminaries: DeepWalk . . . 132
8.3 Binary Network Embedding . . . 133
8.3.1 The Optimization Problem . . . 133
8.3.2 Solving the Optimization Problem . . . 136
8.4 Experiments . . . 139
8.4.1 Datasets . . . 140
8.4.2 Baseline Methods . . . 141
8.4.3 Experimental Settings . . . 143
8.4.4 Evaluation Metrics . . . 144
8.4.5 Similarity Search Results . . . 144
8.4.6 A Case Study on Relevant Paper Search . . . 148
8.4.7 Comparison of Memory Usage . . . 148

8.4.8 Experiments on Search Scalability 150
8.4.9 Comparison of Embedding Learning Time 151
8.4.10 Experiments on Parameter Sensitivity 152
8.5 Conclusion 153

9 Conclusion and Future Work **154**
9.1 Conclusion 154
9.2 Future Work 155

References **157**

List of figures

3.1 A conceptual view of network embedding. Nodes in (a) are indexed using their ID and color coded based on their community information. The network representation learning in (b) transforms all nodes into a two-dimensional vector space, such that nodes with structural proximity are close to each other in the new embedding space. 26

4.1 A toy example of information network. Our proposed HSCA algorithm utilizes all three information sources (*i.e.*, homophily, structural context, and node content), for learning useful network representations. 34
4.2 Convergence of the objective function 49
4.3 Micro-F_1 values with respect to different values of k, μ and λ 50
4.4 Visualization of network representations learned by different algorithms 50

5.1 Node distributions with respect to node degree and the number of node attributes on the Flickr network. 55
5.2 The scatter plot of node content canonical variable and network structure canonical variable. 56
5.3 The working mechanism of the proposed attri2vec algorithm. A transformation $f(\cdot)$ guided by network structure is performed from the original node attribute space to seek a structure-aware attribute subspace, where node attributes and network structure can better compliment each other in a more consistent way towards learning high-quality node representations. 57

5.4 The architecture of attri2vec. For each node context pair (v_i, v_j), attri2vec learns node representations by modeling $\Pr(v_j|v_i)$. attri2vec firstly constructs node representations in the hidden layer by performing a linear or non-linear transformation on v_i's content attributes, then uses the hidden layer representation to predict the probability $\Pr(v_j|v_i)$ with softmax. 61
5.5 Parameter sensitivity study of the algorithm performance (Micro-F_1) in terms of (a): the maximum number of iterations, (b): the window size of the random walks t, and (c): embedding dimension d. 78

6.1 The model architecture of SINE. For each node v_i, SINE learns its representation by using it to predict its context node v_j and its observable attribute a_j so that nodes sharing similar context nodes or similar observed attributes are embedded closely in the new vector space. In this way, the incomplete structure and node attribute information is utilized flexibly to learn informative node representations. 84
6.2 Parameter sensitivity 101
6.3 The comparison of running time of different network embedding algorithms on the log scale 101

7.1 A toy example demonstrating DMF objectives. Given a citation network composed of papers from two categories, Genetic Algorithms (GA) and Neutral Networks (NN), DMF aims to learn informative and discriminative node representations that combine network structure, node content, and sparse node labels to maximally separate nodes into different categorizes with minimum loss, as shown in (b). 107
7.2 Convergence of the objective function 122
7.3 F_1 values with respect to different k values 123
7.4 F_1 values with respect to different μ, λ_1 and λ_2 values 123
7.5 Visualization of learned representations 124

8.1 The model architecture of BinaryNE. For each node v_i, BinaryNE learns its binary representation by using it to predict its context node v_j and its attribute a_j. 134

8.2 Query time with varying $|\mathcal{V}|$ and d 151
8.3 The time consumed by different network embedding methods for learning node representations 152
8.4 The sensitivity of BinaryNE with parameters: the number of iterations, the dimension of learned embeddings d, and the window size t 152

List of tables

1.1 Thesis structure 12

3.1 A summary of common notations 23
3.2 Summary of nine real-world networks 27

4.1 A summary of network embedding algorithms based on the information sources used. 35
4.2 Summary of Four Real-world Networks 43
4.3 Classification Results on Cora 46
4.4 Classification Results on Citeseer 46
4.5 Classification Results on PubMed 47
4.6 Classification Results on Wikipedia 47

5.1 Summary of Four Real-world Networks 65
5.2 Node Classification Results on Citeseer 69
5.3 Node Classification Results on DBLP 69
5.4 Node Classification Results on Facebook 70
5.5 Node Classification Results on Flickr 70
5.6 Node Clustering Results on DBLP 72
5.7 Node Clustering Results on Facebook 72
5.8 Performance Comparison between GD and SGD on Facebook 73
5.9 Performance Comparison of Shallow and Deep Mapping on Citeseer . . 74
5.10 Performance Comparison of Shallow and Deep Mapping on DBLP . . . 74
5.11 A Summary of Time Stamped DBLP Subgraphs 75
5.12 Node Classification Results for Out-of-sample nodes on DBLP 76
5.13 Operators to Construct Edge Features 76

5.14 AUC Values (%) for Predicting the Links of Out-of-sample Nodes on DBLP 77

6.1 Summary of Four Real-world Networks 90
6.2 Node Classification Results on Cora 94
6.3 Node Classification Results on Citeseer 95
6.4 Node Classification Results on DBLP(Subgraph) 96
6.5 Node Clustering Results on Cora 97
6.6 Operators to Construct Edge Features 98
6.7 Heuristic scores for predicting the link between node pair (v_i, v_j) with their direct neighbor sets $\mathcal{N}(v_i)$ and $\mathcal{N}(v_j)$ 98
6.8 AUC Values for Link Prediction on Cora 99
6.9 AUC Values for Link Prediction on DBLP(Full) 100

7.1 Summary of three real-world networks 118
7.2 Average F_1 (%) values on Cora 121
7.3 Average F_1 (%) values on Citeseer 121
7.4 Average F_1 (%) values on PubMed 121

8.1 Summary of six real-world networks 140
8.2 Similarity search results on Cora 145
8.3 Similarity search results on Citeseer 145
8.4 Similarity search results on BlogCatalog 146
8.5 Similarity search results on Flickr 146
8.6 Similarity search results on DBLP(Subgraph) 147
8.7 Top-5 relevant paper search on DBLP 149
8.8 The memory usage of DeepWalk, NetHash and BinaryNE embeddings . 149